K 6

```
FFFFFFFFFFFFFFF     OOOOOOOOO     RRRRRRRRRRRR     RRRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
FFFFFFFFFFFFFFF     OOOOOOOOO     RRRRRRRRRRRR     RRRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
FFFFFFFFFFFFFFF     OOOOOOOOO     RRRRRRRRRRRR     RRRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
FFF              OOO         OOO  RRR         RRR  RRR         RRR        TTT        LLL
FFF              OOO         OOO  RRR         RRR  RRR         RRR        TTT        LLL
FFF              OOO         OOO  RRR         RRR  RRR         RRR        TTT        LLL
FFF              OOO         OOO  RRR         RRR  RRR         RRR        TTT        LLL
FFF              OOO         OOO  RRR         RRR  RRR         RRR        TTT        LLL
FFFFFFFFFFFF     OOO         OOO  RRRRRRRRRRRR     RRRRRRRRRRRR           TTT        LLL
FFFFFFFFFFFF     OOO         OOO  RRRRRRRRRRRR     RRRRRRRRRRRR           TTT        LLL
FFFFFFFFFFFF     OOO         OOO  RRRRRRRRRRRR     RRRRRRRRRRRR           TTT        LLL
FFF              OOO         OOO  RRR   RRR        RRR   RRR              TTT        LLL
FFF              OOO         OOO  RRR   RRR        RRR   RRR              TTT        LLL
FFF              OOO         OOO  RRR   RRR        RRR   RRR              TTT        LLL
FFF              OOO         OOO  RRR      RRR     RRR      RRR           TTT        LLL
FFF              OOO         OOO  RRR      RRR     RRR      RRR           TTT        LLL
FFF              OOO         OOO  RRR      RRR     RRR      RRR           TTT        LLL
FFF                 OOOOOOOOO     RRR         RRR  RRR         RRR        TTT        LLLLLLLLLLLLLLL
FFF                 OOOOOOOOO     RRR         RRR  RRR         RRR        TTT        LLLLLLLLLLLLLLL
FFF                 OOOOOOOOO     RRR         RRR  RRR         RRR        TTT        LLLLLLLLLLLLLLL
```

J 15

```
**FILE**ID**FORDELETE
```

```
FFFFFFFFFF    OOOOOO    RRRRRRRR    DDDDDDDD    EEEEEEEEEE  LL          EEEEEEEEEE  TTTTTTTTTT  EEEEEEEEEE
FFFFFFFFFF    OOOOOO    RRRRRRRR    DDDDDDDD    EEEEEEEEEE  LL          EEEEEEEEEE  TTTTTTTTTT  EEEEEEEEEE
FF          OO      OO  RR      RR  DD      DD  EE          LL          EE              TT      EE
FF          OO      OO  RR      RR  DD      DD  EE          LL          EE              TT      EE
FF          OO      OO  RR      RR  DD      DD  EE          LL          EE              TT      EE
FF          OO      OO  RR      RR  DD      DD  EE          LL          EE              TT      EE
FFFFFFFF    OO      OO  RRRRRRRR    DD      DD  EEEEEEEE    LL          EEEEEEEE        TT      EEEEEEEE
FFFFFFFF    OO      OO  RRRRRRRR    DD      DD  EEEEEEEE    LL          EEEEEEEE        TT      EEEEEEEE
FF          OO      OO  RR  RR      DD      DD  EE          LL          EE              TT      EE
FF          OO      OO  RR  RR      DD      DD  EE          LL          EE              TT      EE
FF          OO      OO  RR    RR    DD      DD  EE          LL          EE              TT      EE              ....
FF          OO      OO  RR    RR    DD      DD  EE          LL          EE              TT      EE              ....
FF            OOOOOO    RR      RR  DDDDDDDD    EEEEEEEEEE  LLLLLLLLLL  EEEEEEEEEE      TT      EEEEEEEEEE      ....
FF            OOOOOO    RR      RR  DDDDDDDD    EEEEEEEEEE  LLLLLLLLLL  EEEEEEEEEE      TT      EEEEEEEEEE      ....

LL            IIIIII      SSSSSSSS
LL            IIIIII      SSSSSSSS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II        SSSSSS
LL              II        SSSSSS
LL              II              SS
LL              II              SS
LL              II              SS
LL              II              SS
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

```
0 MODULE FOR$DELETE (      ! DELETE statement processor
0                 IDENT = '1-002'          ! Edit: SBL1002
0                 ) =
1 BEGIN
1
1 !
1 !****************************************************************************
1 !*                                                                          *
1 !*   COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                *
1 !*   DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                 *
1 !*   ALL RIGHTS RESERVED.                                                   *
1 !*                                                                          *
1 !*   THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  *
1 !*   ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE  *
1 !*   INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER  *
1 !*   COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  *
1 !*   OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY  *
1 !*   TRANSFERRED.                                                           *
1 !*                                                                          *
1 !*   THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE  *
1 !*   AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT  *
1 !*   CORPORATION.                                                           *
1 !*                                                                          *
1 !*   DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS  *
1 !*   SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                *
1 !*                                                                          *
1 !*                                                                          *
1 !****************************************************************************
1 !
1
1 !++
1 ! FACILITY: FORTRAN Language Support Library
1 !
1 ! ABSTRACT:
1 !
1 !       Contains routines to implement FORTRAN DELETE for relative
1 !       and indexed organization files.
1 !
1 ! ENVIRONMENT:  User mode, AST reentrant
1 !
1 ! AUTHOR: Steven B. Lionel, CREATION DATE: 14-May-1979
1 !
1 ! EDIT HISTORY:
1 !
1 ! 1-001 - Original.  SBL 14-May-1979
1 ! 1-002 - Move ACTUALCOUNT declaration inside routine.  SBL 16-June-1982
1 !--
```

```
;    49            0048  1 !
;    50            0049  1 ! SWITCHES:
;    51            0050  1 !
;    52            0051  1
;    53            0052  1 SWITCHES ADDRESSING_MODE (EXTERNAL = GENERAL, NONEXTERNAL = WORD_RELATIVE);
;    54            0053  1
;    55            0054  1 !
;    56            0055  1 ! LINKAGES:
;    57            0056  1 !
;    58            0057  1
;    59            0058  1 REQUIRE 'RTLIN:OTSLNK';                               ! Define all linkages
;    60            0487  1
;    61            0488  1 !
;    62            0489  1 ! TABLE OF CONTENTS:
;    63            0490  1 !
;    64            0491  1
;    65            0492  1 FORWARD ROUTINE
;    66            0493  1         FOR$DELETE : CALL_CCB,                       ! Sequential access delete entry
;    67            0494  1         FOR$DELETE_D : CALL_CCB;                     ! Direct access delete entry
;    68            0495  1
;    69            0496  1 !
;    70            0497  1 ! INCLUDE FILES:
;    71            0498  1 !
;    72            0499  1
;    73            0500  1 LIBRARY 'RTLSTARLE';                                 ! STARLET library for macros and symbols
;    74            0501  1 REQUIRE 'RTLML:FORERR';                              ! FORTRAN error numbers
;    75            0569  1 REQUIRE 'RTLML:OTSLUB';                              ! Logical Unit Block Definitions
;    76            0709  1 REQUIRE 'RTLML:OTSISB';                              ! Inter-statement Block Definitions
;    77            0877  1 REQUIRE 'RTLIN:RTLPSECT';                            ! P-SECT declaration macros
;    78            0972  1 REQUIRE 'RTLML:FORPAR';                              ! Inter-module parameters
;    79            0995  1
;    80            0996  1 !
;    81            0997  1 ! MACROS:
;    82            0998  1 !
;    83            0999  1
;    84            1000  1 !
;    85            1001  1 ! EQUATED SYMBOLS:
;    86            1002  1 !
;    87            1003  1
;    88            1004  1 !
;    89            1005  1 ! PSECT DEFINITIONS:
;    90            1006  1 !
;    91            1007  1
;    92            1008  1 DECLARE_PSECTS (FOR);
;    93            1009  1
;    94            1010  1 !
;    95            1011  1 ! OWN STORAGE:
;    96            1012  1 !
;    97            1013  1
;    98            1014  1 !
;    99            1015  1 ! EXTERNAL REFERENCES:
;   100            1016  1 !
;   101            1017  1
;   102            1018  1 EXTERNAL ROUTINE
;   103            1019  1         FOR$$IOSTAT_HND,                             ! IOSTAT error handler
;   104            1020  1         FOR$$SIGNAL_STO : NOVALUE,                   ! Signal_stop routine
;   105            1021  1         FOR$$SIG_NO_LUB : NOVALUE,                   ! Signal without LUB
;  106         1022  1          FOR$$CB_PUSH : JSB_CB_PUSH NOVALUE,       ! Push a CCB
;  107         1023  1          FOR$$CB_POP : JSB_CB_POP NOVALUE;         ! Pop a CCB
```

```
1 GLOBAL ROUTINE FOR$DELETE (
1         UNIT,                       ! Unit number on which to delete
1         ERR_EQL                     ! 1 if ERR= or IOSTAT= present
1         ) : CALL_CCB =
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 ! FOR$DELETE deletes the current record of the indexed or relative organization file
1 ! open on unit UNIT.
1 !
1 ! CALLING SEQUENCE:
1 !
1 !       iostat.wl.v = FOR$DELETE (unit.rl.v [, err_eql.rlu.v])
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !       unit            - The unit number on which to delete the record.
1 !                         There must be a ''current record'' on this unit.
1 !       err_eql         - If absent or zero, all errors are signalled.
1 !                       - If 1, errors return an IOSTAT error code value.
1 !
1 ! IMPLICIT INPUTS:
1 !
1 !       CCB
1 !
1 ! IMPLICIT OUTPUTS:
1 !
1 !       NONE
1 !
1 ! ROUTINE VALUE:
1 !
1 !       An IOSTAT value.
1 !
1 ! SIDE EFFECTS:
1 !
1 !       SIGNAL_STOP's:
1 !       FOR$_WRIREAFIL - Write to read-only file
1 !       FOR$_NO_CURREC - No current record
1 !       FOR$_DELERR    - Delete error
1 !
1 !--
1
2     BEGIN
2
2     GLOBAL REGISTER
2         CCB = 11: REF BLOCK [, BYTE];
2
2     LOCAL
2         L_UNWIND_ACTION : VOLATILE,
2         L_ERR_EQL_PRES : VOLATILE,
2         STATUS;
2
2     BUILTIN
2         ACTUALCOUNT;
2
2     ENABLE
```

```
    FOR$$IOSTAT_HND (L_UNWIND_ACTION, L_ERR_EQL_PRES);

!+
! Determine if ERR_EQL is present.
!-

IF ACTUALCOUNT () GTR 1
THEN
    L_ERR_EQL_PRES = .ERR_EQL
ELSE
    L_ERR_EQL_PRES = 0;

!+
! Unwind action is NO-OP (no LUB yet)
!-

L_UNWIND_ACTION= FOR$K_UNWINDNOP;

!+
! Get a LUB for this unit.  On return, CCB points to the
! current control block.
!-

FOR$$CB_PUSH (.UNIT, LUB$K_LUN_MIN);

!+
! Unwind action is now to POP the CCB.
!-

L_UNWIND_ACTION = FOR$K_UNWINDPOP;

!+
! If file is not indexed or relative organization, or is direct access,
! signal error FOR$_DELERR.
!-

IF NOT .CCB [LUB$V_NOTSEQORG] OR .CCB [LUB$V_DIRECT]
THEN
    FOR$$SIGNAL_STO (FOR$K_DELERR);

!+
! If file is read-only, signal error FOR$_WRIREAFIL.
!-

IF .CCB [LUB$V_READ_ONLY]
THEN
    FOR$$SIGNAL_STO (FOR$K_WRIREAFIL);

!+
! Try to delete the current record.  If we get an error, signal it.
!-

IF NOT $DELETE (RAB=.CCB)
THEN
    BEGIN
    WHILE .CCB [RAB$L_STS] EQL RMS$_RSA DO
        BEGIN
```

```
;  223          1138  4             $WAIT (RAB=.CCB);
;  224          1139  4             $DELETE (RAB=.CCB);
;  225          1140  3             END;
;  226          1141  3         IF NOT .CCB [RAB$L_STS]
;  227          1142  3         THEN
;  228          1143  3              FOR$$SIGNAL_STO (
;  229          1144  3                 SELECTONEU .CCB [RAB$L_STS] OF
;  230          1145  3                     SET
;  231          1146  3                     [RMS$_CUR, RMS$_RNL] : FOR$K_NO_CURREC;
;  232          1147  3                     [OTHERWISE] : FOR$K_DELERR;
;  233          1148  3                     TES);
;  234          1149  2         END;
;  235          1150  2
;  236          1151  2     !+
;  237          1152  2     ! Return I/O system to previous state
;  238          1153  2     !-
;  239          1154  2
;  240          1155  2     FOR$$CB_POP ();
;  241          1156  2
;  242          1157  2     RETURN 0;           ! Success IOSTAT value
;  243          1158  1     END;


                                                                          .TITLE  FOR$DELETE
                                                                          .IDENT  \1-002\

                                                                          .EXTRN  FOR$$IOSTAT_HND
                                                                          .EXTRN  FOR$$SIGNAL_STO
                                                                          .EXTRN  FOR$$SIG_NO_LUB
                                                                          .EXTRN  FOR$$CB_PUSH, FOR$$CB_POP
                                                                          .EXTRN  SYS$DELETE, SYS$WAIT

                                                                          .PSECT  _FOR$CODE,NOWRT,  SHR,  PIC,2

                                                       081C  00000        .ENTRY  FOR$DELETE, Save R2,R3,R4,R11           ; 1024
                                       54 00000000G  00  9E  00002        MOVAB   SYS$DELETE, R4                          ;
                                       53 00000000G  00  9E  00009        MOVAB   FOR$$SIGNAL_STO, R3                     ;
                                       5E            04  C2  00010        SUBL2   #4, SP                                  ;
                                                     7E  D4  00013        CLRL    L_ERR_EQL_PRES                          ; 1067
                                                04   AE  D4  00015        CLRL    L_UNWIND_ACTION                         ;
                                       6D     0088   CF  DE  00018        MOVAL   12$, (FP)                               ;
                                       01            6C  91  0001D        CMPB    (AP), #1                                ; 1087
                                                     06  1B  00020        BLEQU   1$                                      ;
                                       6E       08   AC  D0  00022        MOVL    ERR_EQL, L_ERR_EQL_PRES                 ; 1089
                                                     02  11  00026        BRB     2$                                      ;
                                                     6E  D4  00028 1$:    CLRL    L_ERR_EQL_PRES                          ; 1091
                                  04   AE            01  D0  0002A 2$:    MOVL    #T, L_UNWIND_ACTION                     ; 1097
                                                     50  D4  0002E        CLRL    R0                                      ; 1104
                                       52       04   AC  D0  00030        MOVL    UNIT, R2                                ;
                                          00000000G  00  16  00034        JSB     FOR$$CB_PUSH                            ;
                                                04   AE  D4  0003A        CLRL    L_UNWIND_ACTION                         ; 1110
                         05       A1   AB            03  E1  0003D        BBC     #3, -95(CCB), 3$                        ; 1117
                         05       FC   AB            04  E1  00042        BBC     #4, -4(CCB), 4$                         ;
                                                     37  DD  00047 3$:    PUSHL   #55                                     ; 1119
                                       63            01  FB  00049        CALLS   #1, FOR$$SIGNAL_STO                     ;
                         05       FC   AB            02  E1  0004C 4$:    BBC     #2, -4(CCB), 5$                         ; 1125
                                                     2F  DD  00051        PUSHL   #47                                     ; 1127
```

```
          63          01 FB 00053         CALLS   #1, FOR$$SIGNAL_STO                 ;
                      5B DD 00056 5$:     PUSHL   CCB                                 ; 1133
          64          01 FB 00058         CALLS   #1, SYS$DELETE                      ;
          3D          50 E8 0005B         BLBS    R0, 11$                             ;
000182DA  8F      08  AB D1 0005E 6$:     CMPL    8(CCB), #99034                      ; 1136
                      10 12 00066         BNEQ    7$                                  ;
                      5B DD 00068         PUSHL   CCB                                 ; 1138
00000000G 00          01 FB 0006A         CALLS   #1, SYS$WAIT                        ;
                      5B DD 00071         PUSHL   CCB                                 ; 1139
          64          01 FB 00073         CALLS   #1, SYS$DELETE                      ;
                      E6 11 00076         BRB     6$                                  ; 1136
          1F      08  AB E8 00078 7$:     BLBS    8(CCB), 11$                         ; 1141
          50      08  AB D0 0007C         MOVL    8(CCB), R0                          ; 1144
000181A0  8F          50 D1 00080         CMPL    R0, #98720                          ; 1146
                      09 13 00087         BEQL    8$                                  ;
000184B4  8F          50 D1 00089         CMPL    R0, #99508                          ;
                      04 12 00090         BNEQ    9$                                  ;
                      35 DD 00092 8$:     PUSHL   #53                                 ;
                      02 11 00094         BRB     10$                                 ;
                      37 DD 00096 9$:     PUSHL   #55                                 ; 1147
          63          01 FB 00098 10$:    CALLS   #1, FOR$$SIGNAL_STO                 ; 1144
            00000000G 00 16 0009B 11$:    JSB     FOR$$CB_POP                         ; 1155
                      50 D4 000A1         CLRL    R0                                  ; 1157
                         04 000A3         RET                                         ; 1158
                       0000 000A4 12$:    .WORD   Save nothing                        ; 1067
          50      08  AC D0 000A6         MOVL    8(AP), R0                           ;
          50      04  A0 D0 000AA         MOVL    4(R0), R0                           ;
                  F8  A0 9F 000AE         PUSHAB  L_ERR_EQL_PRES                      ;
                  FC  A0 9F 000B1         PUSHAB  L_UNWIND_ACTION                     ;
                      02 DD 000B4         PUSHL   #2                                  ;
                      5E DD 000B6         PUSHL   SP                                  ;
          7E      04  AC 7D 000B8         MOVQ    4(AP), -(SP)                        ;
00000000G 00          03 FB 000BC         CALLS   #3, FOR$$IOSTAT_HND                 ;
                         04 000C3         RET                                         ;

; Routine Size:  196 bytes,    Routine Base:  _FOR$CODE + 0000
```

```
1 GLOBAL ROUTINE FOR$DELETE_D (
1         UNIT,                             ! Unit to delete on
1         REC_NO,                           ! Record number to delete
1         ERR_EQL                           ! 1 if ERR= or IOSTAT= specified
1         ) : CALL_CCB =
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 ! Deletes the specified record on a relative organization file
1 ! opened for direct access.
1 !
1 ! CALLING SEQUENCE:
1 !
1 !       iostat.wl.v = FOR$DELETE_D (unit.rl.v, rec_no.rl.v [, err_eql.r;u.v])
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !       unit                        - Unit number to delete on
1 !       rec_no                      - Record number to delete
1 !       err_eql                     - If present and 1, return IOSTAT
1 !                                     values for errors.
1 !
1 ! IMPLICIT INPUTS:
1 !
1 !       CCB
1 !       LUB$L_LOG_RECNO
1 !       LUB$V_FIND_LAST             ! On if FIND was last operation
1 !
1 ! IMPLICIT OUTPUTS:
1 !
1 !       LUB$V_FIND_LAST             ! Cleared
1 !
1 ! ROUTINE VALUE:
1 !
1 !       An IOSTAT value.
1 !
1 ! SIDE EFFECTS:
1 !
1 !       SIGNAL_STOPs:
1 !       FOR$_OPEDEFREQ - Open or define file required for direct or keyed access
1 !       FOR$_WRIREAFIL - Write to read-only file
1 !       FOR$_RECNUMOUT - Record number out of range
1 !       FOR$_ATTACCNON - Attempt to access non-existent record
1 !       FOR$_SPERECLOC - Specified record locked
1 !       FOR$_DELERR    - Delete error
1 !
1 !--
1
2    BEGIN
2
2    GLOBAL REGISTER
2         CCB = 11: REF BLOCK [, BYTE];
2
2    LOCAL
2         L_UNWIND_ACTION : VOLATILE,
2         L_ERR_EQL_PRES : VOLATILE;
```

```
BUILTIN
    ACTUALCOUNT;

ENABLE
    FOR$$IOSTAT_HND (L_UNWIND_ACTION, L_ERR_EQL_PRES);

!+
! Determine if ERR_EQL is present.
!-

IF ACTUALCOUNT () GTR 2
THEN
    L_ERR_EQL_PRES = .ERR_EQL
ELSE
    L_ERR_EQL_PRES = 0;

!+
! Unwind action is NO-OP (no LUB yet)
!-

L_UNWIND_ACTION= FOR$K_UNWINDNOP;

!+
! Get a LUB for this unit.  On return, CCB points to the
! current control block.
!-

FOR$$CB_PUSH (.UNIT, LUB$K_LUN_MIN);

!+
! Unwind action is now to POP the CCB.
!-

L_UNWIND_ACTION = FOR$K_UNWINDPOP;

!+
! If file is not relative organization, signal error FOR$_DELERR.
!-

IF .CCB [LUB$B_ORGAN] NEQU LUB$K_ORG_RELAT
THEN
    FOR$$SIGNAL_STO (FOR$K_DELERR);

!+
! If file is read-only, signal error FOR$_WRIREAFIL.
!-

IF .CCB [LUB$V_READ_ONLY]
THEN
    FOR$$SIGNAL_STO (FOR$K_WRIREAFIL);

!+
! If file is not direct access, signal error FOR$_OPEDEFREQ.
!-

IF NOT .CCB [LUB$V_DIRECT]
```

```
2      THEN
2          FOR$$SIGNAL_STO (FOR$K_OPEDEFREQ);
2
2      !+
2      ! If the record number passed is illegal, signal FOR$_RECNUMOUT.
2      !-
2
3      IF .REC_NO LEQ 0 OR (.CCB [LUB$L_REC_MAX] NEQ 0 AND .REC_NO GTRU .CCB [LUB$L_REC_MAX])
2      THEN
2          FOR$$SIGNAL_STO (FOR$K_RECNUMOUT);
2
2      !+
2      !   See if the requested record to be deleted may already be locked.
2      !   If it is, we want to avoid doing a $FIND which would unlock the
2      !   record.  To check, we compare REC_NO against (LUB$L_LOG_RECNO - 1)
2      !   (or LUB$L_LOG_RECNO if LUB$V_FIND_LAST is set).  If they are equal,
2      !   the last operation we did may have locked this record.  If it didn't,
2      !   then we don't have a record locked.  Trying to delete the record
2      !   will show if it's locked or not.  If not, we do a $FIND anyway.
2      !-
2
2      IF .REC_NO EQL
3          ( IF .CCB [LUB$V_FIND_LAST]
3          THEN
3              .CCB [LUB$L_LOG_RECNO]
3          ELSE
3              .CCB [LUB$L_LOG_RECNO] - 1)
2      THEN
3          BEGIN
3
3          CCB [LUB$V_FIND_LAST] = 0;              ! Turn off bit
3
3          !+
3          ! We may have it locked.  Try to delete it.
3          !-
3
4          IF NOT $DELETE (RAB=.CCB)
3          THEN
3              WHILE .CCB [RAB$L_STS] EQL RMS$_RSA DO
4                  BEGIN
4                  $WAIT (RAB=.CCB);
4                  $DELETE (RAB=.CCB);
3                  END;
3
3          !+
3          ! If we succeeded, return.
3          !-
3
3          IF .CCB [RAB$L_STS]
3          THEN
4              BEGIN
4              FOR$$CB_POP ();      ! Return I/O to previous state
4              RETURN 0;            ! Success IOSTAT value
3              END;
2          END;
2
2      !+
;  416          1330  2     ! We don't have it locked.  Do a $FIND then a $DELETE.
;  417          1331  2     !-
;  418          1332  2
;  419          1333  2     CCB [LUB$V_FIND_LAST] = 0;
;  420          1334  2
;  421          1335  2     CCB [LUB$L_LOG_RECNO] = .REC_NO;
;  422          1336  3     IF NOT $FIND (RAB=.CCB)
;  423          1337  2     THEN
;  424          1338  3         BEGIN
;  425          1339  3         WHILE .CCB[RAB$L_STS] EQL RMS$_RSA DO
;  426          1340  4             BEGIN
;  427          1341  4         $WAIT (RAB=.CCB);
;  428          1342  5         $FIND (RAB=.CCB)
;  429          1343  3         END;
;  430          1344  3         IF NOT .CCB [RAB$L_STS] THEN FOR$$SIGNAL_STO (
;  431          1345  3         SELECTONEU .CCB [RAB$L_STS] OF
;  432          1346  3             SET
;  433          1347  3             [RMS$_RLK] : FOR$K_SPERECLOC;
;  434          1348  3             [RMS$_RNF, RMS$_EOF] : FOR$K_ATTACCNON;
;  435          1349  3             [OTHERWISE] : FOR$K_DELERR;
;  436          1350  3             TES);
;  437          1351  2         END;
;  438          1352  2
;  439          1353  2     !+
;  440          1354  2     ! Try to delete the current record.  If we get an error, signal it.
;  441          1355  2     !-
;  442          1356  2
;  443          1357  3     IF NOT $DELETE (RAB=.CCB)
;  444          1358  2     THEN
;  445          1359  3         BEGIN
;  446          1360  3         WHILE .CCB [RAB$L_STS] EQL RMS$_RSA DO
;  447          1361  4             BEGIN
;  448          1362  4             $WAIT (RAB=.CCB);
;  449          1363  4             $DELETE (RAB=.CCB);
;  450          1364  3             END;
;  451          1365  3         IF NOT .CCB [RAB$L_STS]
;  452          1366  3         THEN
;  453          1367  3             FOR$$SIGNAL_STO (FOR$K_DELERR);
;  454          1368  2         END;
;  455          1369  2
;  456          1370  2     !+
;  457          1371  2     ! Return I/O system to previous state
;  458          1372  2     !-
;  459          1373  2
;  460          1374  2     FOR$$CB_POP ();
;  461          1375  2
;  462          1376  2     RETURN 0;               ! Success IOSTAT value
;  463          1377  1     END;


                                                  .EXTRN  SYS$FIND

                              087C 00000          .ENTRY  FOR$DELETE_D, Save R2,R3,R4,R5,R6,R11      ; 1159
          56 00000000G  00   9E 00002          MOVAB   SYS$FIND, R6                                ;
          55 00000000G  00   9E 00009          MOVAB   SYS$WAIT, R5                                ;
          54 00000000G  00   9E 00010          MOVAB   SYS$DELETE, R4                              ;
```

```
                 53 00000000G  00  9E 00017        MOVAB   FOR$$SIGNAL_STO, R3                          ;
                 5E            04  C2 0001E        SUBL2   #4, SP                                       ;
                               7E  D4 00021        CLRL    L_ERR_EQL_PRES                               ; 1208
                        04     AE  D4 00023        CLRL    L_UNWIND_ACTION                              ;
                 6D   0117     CF  DE 00026        MOVAL   23$, (FP)                                    ;
                 02            6C  91 0002B        CMPB    (AP), #2                                     ; 1227
                               06  1B 0002E        BLEQU   1$                                           ;
                 6E     0C     AC  D0 00030        MOVL    ERR_EQL, L_ERR_EQL_PRES                      ; 1229
                               02  11 00034        BRB     2$                                           ;
                               6E  D4 00036 1$:    CLRL    L_ERR_EQL_PRES                               ; 1231
          04     AE            01  D0 00038 2$:    MOVL    #1, L_UNWIND_ACTION                          ; 1237
                               50  D4 0003C        CLRL    R0                                           ; 1244
                 52     04     AC  D0 0003E        MOVL    UNIT, R2                                     ;
                  00000000G    00  16 00042        JSB     FOR$$CB_PUSH                                 ;
                        04     AE  D4 00048        CLRL    L_UNWIND_ACTION                              ; 1250
                 02     C4     AB  91 0004B        CMPB    -60(CCB), #2                                 ; 1256
                               05  13 0004F        BEQL    3$                                           ;
                               37  DD 00051        PUSHL   #55                                          ; 1258
                 63            01  FB 00053        CALLS   #1, FOR$$SIGNAL_STO                          ;
   05     FC     AB            02  E1 00056 3$:    BBC     #2, -4(CCB), 4$                              ; 1264
                               2F  DD 0005B        PUSHL   #47                                          ; 1266
                 63            01  FB 0005D        CALLS   #1, FOR$$SIGNAL_STO                          ;
   05     FC     AB            04  E0 00060 4$:    BBS     #4, -4(CCB), 5$                              ; 1272
                               1A  DD 00065        PUSHL   #26                                          ; 1274
                 63            01  FB 00067        CALLS   #1, FOR$$SIGNAL_STO                          ;
                 52     08     AC  D0 0006A 5$:    MOVL    REC_NO, R2                                   ; 1280
                               0B  15 0006E        BLEQ    6$                                           ;
                        E4     AB  D5 00070        TSTL    -28(CCB)                                     ;
                               0B  13 00073        BEQL    7$                                           ;
          E4     AB            52  D1 00075        CMPL    R2, -28(CCB)                                 ;
                               05  1B 00079        BLEQU   7$                                           ;
                               19  DD 0007B 6$:    PUSHL   #25                                          ; 1282
                 63            01  FB 0007D        CALLS   #1, FOR$$SIGNAL_STO                          ;
   06     A0     AB            03  E1 00080 7$:    BBC     #3, -96(CCB), 8$                             ; 1295
                 50     E0     AB  D0 00085        MOVL    -32(CCB), R0                                 ; 1297
                               05  11 00089        BRB     9$                                           ;
   50     E0     AB            01  C3 0008B 8$:    SUBL3   #1, -32(CCB), R0                             ; 1299
                 50            52  D1 00090 9$:    CMPL    R2, R0                                       ; 1295
                               26  12 00093        BNEQ    12$                                          ;
          A0     AB            08  8A 00095        BICB2   #8, -96(CCB)                                 ; 1303
                               5B  DD 00099        PUSHL   CCB                                          ; 1309
                 64            01  FB 0009B        CALLS   #1, SYS$DELETE                               ;
                 16            50  E8 0009E        BLBS    R0, 11$                                      ;
    000182DA     8F     08     AB  D1 000A1 10$:   CMPL    8(CCB), #99034                               ; 1311
                               0C  12 000A9        BNEQ    11$                                          ;
                               5B  DD 000AB        PUSHL   CCB                                          ; 1313
                 65            01  FB 000AD        CALLS   #1, SYS$WAIT                                 ;
                               5B  DD 000B0        PUSHL   CCB                                          ; 1314
                 64            01  FB 000B2        CALLS   #1, SYS$DELETE                               ;
                               EA  11 000B5        BRB     10$                                          ; 1311
                 7D     08     AB  E8 000B7 11$:   BLBS    8(CCB), 22$                                  ; 1321
          A0     AB            08  8A 000BB 12$:   BICB2   #8, -96(CCB)                                 ; 1333
          E0     AB            52  D0 000BF        MOVL    R2, -32(CCB)                                 ; 1335
                               5B  DD 000C3        PUSHL   CCB                                          ; 1336
                 66            01  FB 000C5        CALLS   #1, SYS$FIND                                 ;
                 46            50  E8 000C8        BLBS    R0, 19$                                      ;
    000182DA     8F     08     AB  D1 000CB 13$:   CMPL    8(CCB), #99034                               ; 1339
```

```
                        0C 12 000D3       BNEQ    14$                           ;
                        5B DD 000D5       PUSHL   CCB                           ; 1341
          65            01 FB 000D7       CALLS   #1, SYS$WAIT                  ;
                        5B DD 000DA       PUSHL   CCB                           ; 1342
          66            01 FB 000DC       CALLS   #1, SYS$FIND                  ;
                        EA 11 000DF       BRB     13$                           ;
          2C         08 AB E8 000E1 14$:  BLBS    8(CCB), 19$                   ; 1344
          50         08 AB D0 000E5       MOVL    8(CCB), R0                    ; 1345
000182AA  8F            50 D1 000E9       CMPL    R0, #98986                    ; 1347
                        04 12 000F0       BNEQ    15$                           ;
                        34 DD 000F2       PUSHL   #52                           ;
                        18 11 000F4       BRB     18$                           ;
0001827A  8F            50 D1 000F6 15$:  CMPL    R0, #98938                    ; 1348
                        09 13 000FD       BEQL    16$                           ;
000182B2  8F            50 D1 000FF       CMPL    R0, #98994                    ;
                        04 12 00106       BNEQ    17$                           ;
                        24 DD 00108 16$:  PUSHL   #36                           ;
                        02 11 0010A       BRB     18$                           ;
                        37 DD 0010C 17$:  PUSHL   #55                           ; 1349
          63            01 FB 0010E 18$:  CALLS   #1, FOR$$SIGNAL_STO           ; 1345
                        5B DD 00111 19$:  PUSHL   CCB                           ; 1357
          64            01 FB 00113       CALLS   #1, SYS$DELETE                ;
          1F            50 E8 00116       BLBS    R0, 22$                       ;
000182DA  8F         08 AB D1 00119 20$:  CMPL    8(CCB), #99034                ; 1360
                        0C 12 00121       BNEQ    21$                           ;
                        5B DD 00123       PUSHL   CCB                           ; 1362
          65            01 FB 00125       CALLS   #1, SYS$WAIT                  ;
                        5B DD 00128       PUSHL   CCB                           ; 1363
          64            01 FB 0012A       CALLS   #1, SYS$DELETE                ;
                        EA 11 0012D       BRB     20$                           ; 1360
          05         08 AB E8 0012F 21$:  BLBS    8(CCB), 22$                   ; 1365
                        37 DD 00133       PUSHL   #55                           ; 1367
          63            01 FB 00135       CALLS   #1, FOR$$SIGNAL_STO           ;
              00000000G 00 16 00138 22$:  JSB     FOR$$CB_POP                   ; 1374
                        50 D4 0013E       CLRL    R0                            ; 1377
                           04 00140       RET                                   ;
                         0000 00141 23$:  .WORD   Save nothing                  ; 1208
          50         08 AC D0 00143       MOVL    8(AP), R0                     ;
          50         04 A0 D0 00147       MOVL    4(R0), R0                     ;
                     F8 A0 9F 0014B       PUSHAB  L_ERR_EQL_PRES                ;
                     FC A0 9F 0014E       PUSHAB  L_UNWIND_ACTION               ;
                        02 DD 00151       PUSHL   #2                            ;
                        5E DD 00153       PUSHL   SP                            ;
          7E         04 AC 7D 00155       MOVQ    4(AP), -(SP)                  ;
00000000G 00            03 FB 00159       CALLS   #3, FOR$$IOSTAT_HND           ;
                           04 00160       RET                                   ;

; Routine Size:  353 bytes,    Routine Base:  _FOR$CODE + 00C4
;  465         1378  1 END                                      !End of module
;  466         1379  0 ELUDOM



;                                       PSECT SUMMARY
;
;       Name                         Bytes                           Attributes
;
;  _FOR$CODE                                 549  NOVEC,NOWRT,  RD ,  EXE,  SHR,  LCL,  REL,  CON,  PIC,ALIGN(2)



;                              Library Statistics
;
;                                              -------- Symbols --------          Pages      Processing
;       File                                   Total    Loaded   Percent          Mapped     Time
;
;  _$255$DUA28:[SYSLIB]STARLET.L32;1            9776        13         0           581          00:01.1




;                                       COMMAND QUALIFIERS

;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACE/LIS=LIS$:FORDELETE/OBJ=OBJ$:FORDELETE MSRC$:FORDELETE/UPDATE=(ENH$:FORDELETE
;       )

; Size:          549 code + 0 data bytes
; Run Time:         00:16.4
; Elapsed Time:     00:55.0
; Lines/CPU Min:     5045
; Lexemes/CPU-Min: 33318
; Memory Used:  185 pages
; Compilation Complete
```

0179 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

COMR50WD
LIS

FORDATEDS
LIS

FORDECOMO
LIS

FORCB
LIS

COMSETST
LIS

FORASSOC
LIS

FORCLOSEF
LIS

FORDATE
LIS

FORCLOSE
LIS

FORDECOMP
LIS

FORDELETE
LIS

COMRAD50
LIS

COMUSEREX
LIS

FORBITOPS
LIS

FORDEFINE
LIS

FORBACKSP
LIS

FORDISPAT
LIS

FORCVTRT
LIS